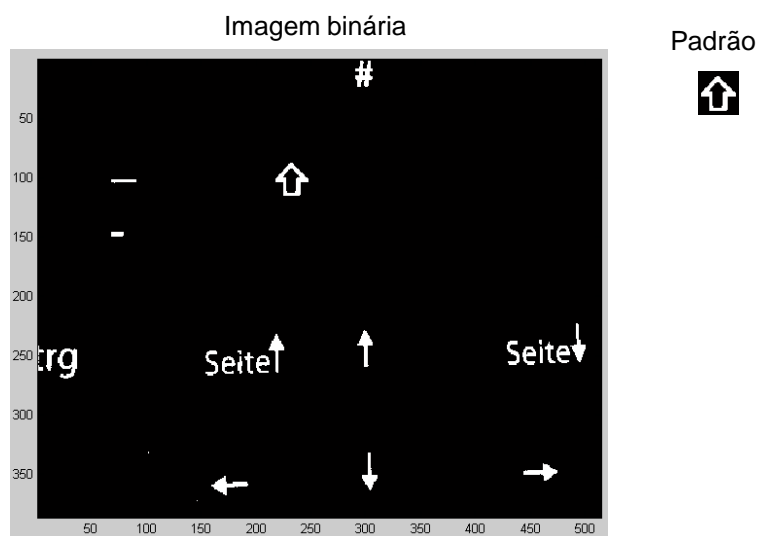


# Programa

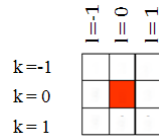
1. Fundamentos de imagens digitais
2. Realce de contraste
3. Filtragem
4. Correlação
5. Segmentação

Problema:

Encontrar a posição de um padrão P na imagem I.



### Operadores convolução e correlação:



Padrão (P) com  
dimensão **mp** linhas e  
**mp** colunas.

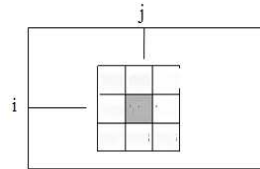


Imagem (I) com **m** linhas  
e **n** colunas.

Convolução: 
$$S(i,j) = \sum_{k=-d1}^{d1} \sum_{l=-d1}^{d1} P(k,l) \cdot I(i-k, j-l)$$

Correlação: 
$$S(i,j) = \sum_{k=-d1}^{d1} \sum_{l=-d1}^{d1} P(k,l) \cdot I(i+k, j+l)$$

Implementação para a posição ( i , j ) :

```
soma = 0;
for k = -d1:d1
for l = -d1:d1
    soma = soma+ P(d2+k, d2+l) * I(i+k, j+l);
end
end
S(i,j) = round(soma);
```

Para imagens em tons de cinza:

- coeficiente de correlação normalizado

$$S(i,j) = \frac{\sum_{k=-d1}^{d1} \sum_{l=-d1}^{d1} P(k,l) \cdot I(i+k, j+l)}{\sqrt{\sum_{k=-d1}^{d1} \sum_{l=-d1}^{d1} P(k,l)^2 \cdot I(i+k, j+l)^2}}$$

Padrão P:  
padrao\_binario.tif

Imagem I:  
teclado\_binario.tif

```
[m,n] = size(I);  
[mp,np] = size(P);  
d1 = floor(mp/2);  
d2=ceil(mp/2);  
I = double(I);  
P = double(P);  
S = zeros(m,n); % imagem de saída  
  
for i = d2:m-d1  
for j = d2:n-d1  
soma = 0;  
for k = -d1:d1  
for l = -d1:d1  
soma = soma+ P(d2+k, d2+l) * I(i+k, j+l);  
end  
end  
S(i,j) = round(soma);  
end  
end  
  
imagesc(S)  
colormap gray
```

Posição onde a correlação é máxima:  
[i j] = find(S == max(max(S)))

Exercício 4: Determine a posição na imagem I onde ocorre o padrão P.

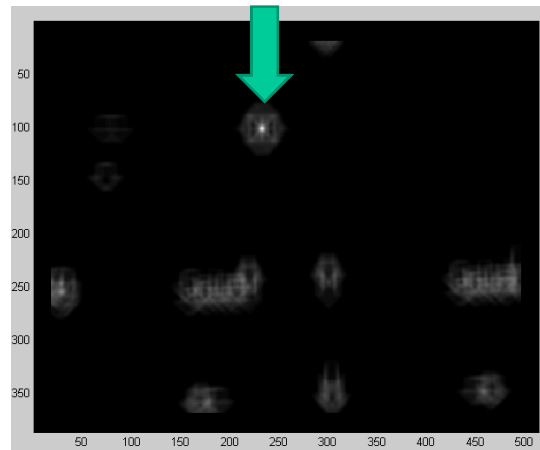
Padrão P:  
padrao\_binario.tif



Imagem I:  
teclado\_binario.tif



Posição de melhor casamento entre o padrão P e a imagem:



% (linha, coluna) do melhor casamento do padrão P com a imagem  
[i j] = find(S == max(max(S)))

#### Exercício 5:

Determine a posição na imagem “teclado\_8bits.tif” onde ocorre o padrão “padrão\_8bits.tif”.


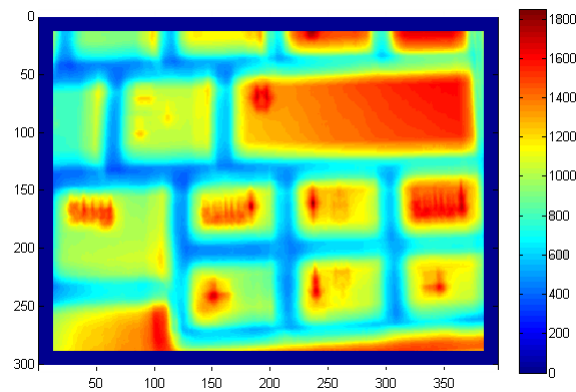
Padrão: 

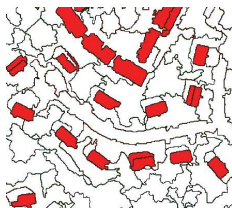
Imagem:





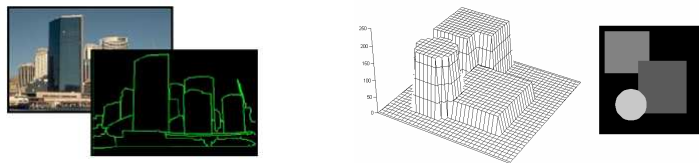
## Segmentação

- A segmentação visa dividir a imagem em regiões ou objetos.
  - O nível de detalhamento na depende da aplicação em questão.



## Abordagens para a segmentação

- As abordagens convencionais para a segmentação de imagens são normalmente buscam detectar **descontinuidades** ou **similaridades** na imagem a partir da análise de níveis de cinza (ou de cores).
  - Descontinuidade -> mudança abrupta nos níveis de cinza.
  - Similaridade -> agrupar pixels que apresentam valores similares para determinado conjunto de características.



## Abordagens para a segmentação :

-> global : é efetuada para a imagem inteira, simultaneamente;

-> local : é analisada a vizinhança e, em geral, é implementada de modo iterativo.

Segmentação **global** da imagem:

-> usa um critério de similaridade/dissimilaridade

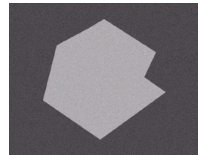
Exemplos:

- Limiarização do histograma
- Segmentação no espaço de cores HSI

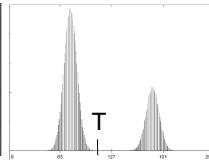
## Limização:

Consiste em especificar um ou mais limiares a partir da análise do histograma da imagem.

Fundamentação: O nível digital (ND) de um objeto se encontra dentro de um intervalo  
 $ND_{\text{minimo}} < ND < ND_{\text{maximo}}$



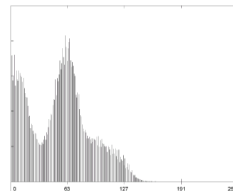
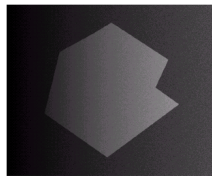
Imagem



Histograma

Analisar cada pixel  $I(i,j)$  tal que  
 se  $I(i,j) > T$  é denominado ponto do objeto,  
 caso contrário, fundo.

$$R(i,j) = \begin{cases} 0 & \text{se } I(i,j) < T \\ 1 & \text{se } I(i,j) \geq T \end{cases}$$



A limiarização é adequada para segmentar esta imagem?

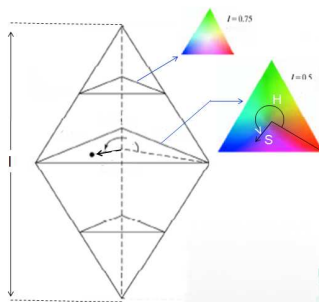
Segmentação no espaço de cores HSI:

Exercício: segmentar objetos azuis.

Imagem "lapis1.tif"



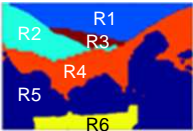
-> espaço de cores HSI:



H	Cor
0°	Vermelho
120°	Verde
240°	Azul
360°	Vermelho

Segmentação orientada a regiões

Seja I a imagem de entrada:



A **segmentação** particiona I em n regiões conexas R1, R2,..., Rn,

tal que:

- a segmentação deve ser completa, isto é, cada pixel deve pertencer a uma região.	$\bigcup_{i=1}^n R_i = I$
- as regiões devem ser disjuntas.	$R_i \cap R_j = \emptyset \quad \forall i = 1, 2, \dots, n$
- Pixels de uma região devem satisfazer uma propriedade comum.	$P(R_i) = \text{VERDADEIRO para } i = 1, 2, \dots, n$
- Regiões adjacentes Ri e Rj não podem ser unidas numa única região.	$P(R_i \cup R_j) = \text{FALSO para } i \neq j$

Uma região consiste de um grupo de pixels conectados e com propriedades similares.

Para se estabelecer se dois pixels estão **conectados** é necessário determinar se eles são adjacentes e se seus níveis de cinza satisfazem a um determinado critério de similaridade.

1

0

1

1

1

0

0

0

0

Vizinhança de um pixel (i,j):

-Vizinhança 4:

I(i-1,j-1)	I(i-1,j)	I(i-1,j+1)
I(i,j-1)	I(i,j)	I(i,j+1)
I(i+1,j-1)	I(i+1,j)	I(i+1,j+1)

-Vizinhança 8:

I(i-1,j-1)	I(i-1,j)	I(i-1,j+1)
I(i,j-1)	I(i,j)	I(i,j+1)
I(i+1,j-1)	I(i+1,j)	I(i+1,j+1)

- Distância entre níveis digitais como medida de similaridade:  
 $D = \text{abs}(ND1 - ND2)$

- Distância euclidiana como medida de similaridade:  
 $D = \sqrt{(ND1 - ND2)^2}$

Exemplo.  
Distância euclidiana entre o ND cujo valor é 2 e o ND cujo valor é 5:

$D = \text{sqrt}((2 - 5)^2)$



Exemplo de segmentação orientada a região:

-Crescimento de regiões:

- > inicia com pixels "semente" representativos de cada região
- > utiliza um critério similaridade
- > é iterativo
  - cada pixel deve ser alocado a uma região

## Crescimento de Regiões

É um método iterativo de segmentação da imagem em regiões.

### Concepção do método:

Iniciando com pontos semente, as regiões são formadas pelo agrupamento de pixels vizinhos que possuem propriedades similares.

O **critério de similaridade** pode ser definido como um limiar de intensidade, textura, cor, etc.



### Algoritmo:

1. Escolher os pontos (DNs) semente.
2. Se os pixels vizinhos satisfizerem o critério de similaridade, então as regiões crescerão.
3. Repetir o passo 2 até que todos os pixels da imagem sejam incluídos numa das regiões.

**Exemplo:**

Segmentar a imagem I em 3 regiões, considerando vizinhança 4 e as sementes:

S1 = 1,  
S2 = 5,  
S3 = 9,

1	1	9	9	9
1	1	9	9	9
5	1	1	9	9
5	5	5	3	9
3	3	3	3	3

*Critério de similaridade:* Se  $\text{abs}(\text{ND}(i,j) - S_k) \leq T$ , então ND(i,j) pertence à região  $R_k$   
*Obs.:* T é um limiar.

1	0	0	9	0
1	1	9	9	9
0	0	0	9	0
5	5	5	0	0
0	5	0	0	0

Iteração 1,  
conectividade 4

1	0	9	9	9
1	1	9	9	9
5	1	0	9	9
5	5	5	0	0
5	5	5	0	0

Iteração 1,  
conectividade 8

**Exemplo: Crescimento de regiões.**

Segmentar a imagem I, considerando a vizinhança-8 e 3 sementes localizadas nas posições :

s1 = I(2,1) ; s2 = I(2,4); s3 = I(4,2)

e limiar T = 2.

Atribuir às regiões, respectivamente, os valores 1, 2 e 3.

1	1	9	9	9
1	1	9	9	9
5	1	1	9	9
5	5	5	4	9
6	6	6	6	4

**Preliminares:**

```
[m,n] = size(I);
R = zeros(m,n); % inicialização da imagem de saída
ns = 3; % num sementes
s = [ 1 9 5 ]; % valor sementes
s_lin = [2 2 4] % local. sementes (linhas)
s_col = [2 4 2] % local. sementes (colunas)
r = [ 1 2 3 ]; % identifica região
T = 2 % limiar T
% atribuindo em R o valor (identificação) de cada região
for i = 1:ns
    R(s_lin(i), s_col(i)) = r(i)
end
```

0	0	0	0	0
0	1	0	2	0
0	0	0	0	0
0	3	0	0	0
0	0	0	0	0

```
it = 1 %Para a primeira iteração:
```

```
for i = 2:m-1
for j = 2:n-1
for k = -1:1
for l = -1:1

% conectividade 8

if R(i,j) ~= 0 & R(i+k,j+l) == 0
    if R(i,j) == r(1) & ( abs( s(1) - I(i+k,j+l) ) <= T )
        R(i+k,j+l) = r(1);
    end
    if R(i,j) == r(2) & ( abs( s(2) - I(i+k,j+l) ) <= T )
        R(i+k,j+l) = r(2);
    end
    if R(i,j) == r(3) & ( abs( s(3) - I(i+k,j+l) ) <= T )
        R(i+k,j+l) = r(3);
    end
end
end
end %end l
end %end k
end
end
```

```
%Para a próxima iteração:
it = it + 1
```

```
%Para a primeira iteração:
```

```
it = 1
```

```
while min(min(R)) == 0
```

```
for i = 2:m-1
for j = 2:n-1
for k = -1:1
for l = -1:1

% conectividade 8

if R(i,j) ~= 0 & R(i+k,j+l) == 0
    if R(i,j) == r(1) & ( abs( s(1) - I(i+k,j+l) ) <= T )
        R(i+k,j+l) = r(1);
    end
    if R(i,j) == r(2) & ( abs( s(2) - I(i+k,j+l) ) <= T )
        R(i+k,j+l) = r(2);
    end
    if R(i,j) == r(3) & ( abs( s(3) - I(i+k,j+l) ) <= T )
        R(i+k,j+l) = r(3);
    end
end
end
end %end l
end %end k
end
end
```

```
%Para a próxima iteração:
```

```
it = it + 1
```

```
end
```

Exercício 2. Crescimento de regiões.

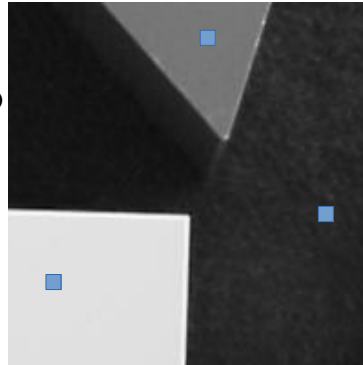
-> Imagem: *blocos.tif*

Segmentar a imagem, considerando a vizinhança-8 e três sementes.

```
% 3 sementes
R = zeros(m,n);
ns = 3; % num sementes
s = [ 30 117 225] ; % valor sementes
r = [ 1 2 3] ; % identifica regiões
s_lin = [92 13 77] % local. sementes (linhas)
s_col = [81 56 28] % local. sementes (colunas)
T = 30 % limiar T

for i = 1:ns
R(s_lin(i), s_col(i)) = r(i)
end

it = 1;
R1 = R(2:m-1,2:n-1);
```



Processamento Digital de Imagens - Primeiro semestre de 2018  
Segundo Trabalho: Transformação geométrica

**Considerando as duas molduras e a imagem fornecida, preencha devidamente uma das molduras com o respectivo quadro.**

Faça um relatório descrevendo os passos e os cálculos realizados.

Apresentar no relatório:

- Tabelas contendo as coordenadas medidas;
- As equações matriciais, explicitando os valores das matrizes utilizadas nos cálculos;
- A imagem resultante do processamento.

Prazo: até o dia 18/06/18.

\\Docs.ufpr.br\haraki\htm\PDI\_1\TP2